Privacy Risks in Trajectory Data Publishing: Reconstructing Private Trajectories from

Emre Kaplan, Thomas B. Pedersen, Erkay Savaş, and Yücel Saygın

Faculty of Engineering & Natural Sciences
Sabanci University, Istanbul, Turkey

Abstract. Location and time information about individuals can be captured through GPS devices, GSM phones, RFID tag readers, and by other similar means. Such data can be pre-processed to obtain trajectories which are sequences of spatio-temporal data points belonging to a moving object. Recently, advanced data mining techniques have been developed for extracting patterns from moving object trajectories to enable applications such as city traffic planning, identification of evacuation routes, trend detection, and many more. However, when special care is not taken, trajectories of individuals may also pose serious privacy risks even after they are de-identified or mapped into other forms. In this paper, we show that an unknown private trajectory can be reconstructed from knowledge of its properties released for data mining, which at first glance may not seem to pose any privacy threats. In particular, we propose a technique to demonstrate how private trajectories can be re-constructed from knowledge of their distances to a bounded set of known trajectories. Experiments performed on real data sets show that the number of known samples is surprisingly smaller than the actual theoretical bounds.

Keywords: Privacy, Spatio-temporal data, trajectories, data mining.

1 Introduction

Information about our location is being collected via an ever-increasing number of devices and by an increasing number of parties, e.g. private companies and public organizations. Phone companies can track our movements via our cellphones. Banks register time and location information for our financial transactions we performed using our credit cards. A growing number of RFID tags are being used to give us access to, e.g., parking spaces or public transportation. Considering the current trend, there is no doubt that the amount of spatio-temporal data being collected will increase drastically in the future. From the point of view of data-analysis, the availability of all this information gives us the

* This work was partially funded by the Information Society Technologies Programme of the European Commission, Future and Emerging Technologies under IST-014915 GeoPKDD project.

I. Lovrek, R.J. Howlett, and L.C. Jain (Eds.): KES 2008, Part II, LNAI 5178, pp. 642–649, 2008.

ability to find new and interesting patterns about how people move in the public space. For instance, such patterns will be useful in solving the growing traffic problems in many metropolitan areas. On the other hand, collection of all these time and location pairs of individuals enables anyone, who observes the data, to reconstruct the movements (the trajectory) of others with a very high precision. There is a growing concern about this serious threat to privacy of individuals whose whereabouts are easily monitored and tracked. Legal and technical aspects of such threats were highlighted at a recent workshop on mobility, data mining, and privacy [3].

In this paper we consider the following scenario: A malicious person wishes to reconstruct the movements (the "target trajectory") of a specific individual. The malicious person does not know the trajectory itself, but only various properties of the trajectory, such as the average speed, a few points visited, or the average distance between the target trajectory and a few trajectories known to the malicious person. We propose a concrete algorithm which can reconstruct the target trajectory from this information.

Despite privacy concerns, many techniques were proposed to mine useful patterns from trajectories. Some of the very recent results are [5,7,8,10] where in [5] the authors mine for temporal patterns of the form $a \rightarrow^t b$ meaning that t is the typical time to travel from location a to location b. Their algorithm needs to know what points of interests the trajectories pass through, and at which time intervals. In [7] the authors give a clustering algorithm which considers sub-trajectories. The main observation is that sub-parts of trajectories may follow interesting common patterns, while the trajectories as a whole may be very different from each other. In [8] authors give a method for finding "hot-routes" in a given road network, which can help us in traffic management.

In all the algorithms mentioned above different properties of the trajectories are needed. Some methods only need the mutual distances between trajectories, some need the exact trajectories, and others only need to know at what times the trajectories pass through certain areas of interest. In this paper, we show how, even very little, information is enough to recover the movement behavior of an individual. In particular we demonstrate how an unknown trajectory can be almost entirely reconstructed from its distance to a few fixed trajectories.

Previous work on spatio-temporal data privacy include anonymization in location based services. Some of the recent work include [9,2]. However, they do not deal with trajectory data. Techniques for trajectory anonymization were recently proposed in [1] but privacy risks after data release were not considered. In another recent work, privacy risks due to distance preserving data transformations were identified [13], however spatio-temporal data was not addressed.

Contributions of this work can be summarized as follows: 1) We demonstrate that trajectories can be reconstructed very precisely with very limited information using relatively simple methods. In particular we show that for a real world dataset of bus trajectories in Athens, we can reconstruct an unknown

trajectory with 1096 sample points by knowing its distance to only 40-50 known trajectories. This is in sharp contrast to the 2193 known distances which would be needed to solve the corresponding system of equations to find the unknown trajectory. 2) We propose a method which can reconstruct trajectories from a very wide range of continuous properties (cf. Section 2); the method of known distances is only a special case. Our method is optimal in the sense that it will eventually find a candidate which exhausts all the information available about the unknown trajectory.

2 Trajectories and Continuous Properties

In their most general form trajectories are paths in space–time. In practice, however, trajectories are collected with GPS devices, or other discrete sampling methods. A discrete trajectory is a polyline represented as a list of sample-points: $T = ((x_1, y_1, t_1), \ldots, (x_n, y_n, t_n))$. We write T_i to represent the ith sample-point (x_i, y_i, t_i). In most of this paper we think of a trajectory as a column-vector in a large vector-space. We use calligraphic letters to refer to the vector representation of a trajectory. The vector representation of a trajectory T is: $\mathscr{T} = (x_1, y_1, t_1, \ldots, x_n, y_n, t_n)^T \in \mathbb{R}^{3n}$. In this case $\mathscr{T}_i$ is the ith element of the vector (i.e. $\mathscr{T}_1 = x_1, \mathscr{T}_2 = y_1, \ldots, \mathscr{T}_{3n} = t_n$).

In this paper we assume that trajectories are 1) are *aligned*[1] and 2) have constant sampling rate ($t_{i+1} - t_i = c$, for some constant c). Algorithms for ensuring these conditions can be found in [6]. In consequence we discard the time component and represent a trajectory as a list of (x, y) coordinates (or a vector in $\mathbb{R}^{2n}$).

A trajectory $\mathscr{T}$ can posses many properties which are of interest in different situations, such as maximum and average speed of a trajectory, closest distance to certain locations, duration of longest "stop", or percentage of time that $\mathscr{T}$ moves "on road". In this work we show how any property of $\mathscr{T}$ which can be expressed as a continuously differentiable function $f : \mathbb{R}^{2n} \to \mathbb{R}$ can be used to reconstruct $\mathscr{T}$. All the examples given above are continuously differentiable properties of $\mathscr{T}$.

The experiments in Section 5 are performed by using an important property of trajectories, namely the distance from an unknown trajectory $\mathscr{T}$ to a fixed trajectory, $\mathscr{T}'$. When using a continuously differentiable norm to compute the distance between $\mathscr{T}$ and $\mathscr{T}'$ we obtain a continuously differentiable property of $\mathscr{T}$; e.g. $\Delta_{\mathscr{T}'}(\mathscr{T}) = d(\mathscr{T}', \mathscr{T})$ is continuously differentiable. Several distance measures for trajectories have been proposed [11], but in the experiments in this paper we focus on Euclidean distance:

$$\|\mathscr{T} - \mathscr{T}'\|_2 = \sqrt{\sum_{i=1}^{2n} |\mathscr{T}_i - \mathscr{T}'_i|^2}, \tag{1}$$

[1] Two trajectories are aligned if they have the same sampling times and the same number of sample points.

3 Reconstructing Trajectories

In this paper we consider how a malicious person can find an unknown trajectory, X, with as little information as possible. Any information we have about X may improve our ability to reconstruct X; a car does not drive in the ocean, and rarely travels at a speed of more than 200 km/h. With a sufficient number of known properties of X, the trajectory can be fully reconstructed. If, for example, $2n$ linear properties of X are known, we have a system of $2n$ linear equations. Solving these $2n$ equations gives us the exact unknown trajectory. The number of linear properties we need to know, however, is at least as large as the number of coordinates in the trajectory itself. If only $m \ll 2n+1$ linear properties are known, the solution will be in a $(2n-m)$-dimensional subspace, at best. When the candidate can only be restricted to a subspace, it can be arbitrarily far away from X. If the known properties are non-linear, finding a solution to the corresponding equations, even if sufficient number of properties is known, may even become infeasible.

As seen from this discussion, a method which can approximate the unknown trajectory with considerably fewer known properties than coordinates is needed. The method presented in the next section is an important step in this direction.

In the rest of this paper we limit our study to information about Euclidean distance between the unknown trajectory and $m \ll 2n+1$ known trajectories, and leave it to future work to include other properties of trajectories. The method we propose in the next section, however, can easily be extended to handle any continuously differentiable property. Thus, the problem addressed in the rest of this paper is as follows: Given m trajectories, $\mathscr{T}_1, \ldots, \mathscr{T}_m$, and m corresponding positive real values δ_i, ε_i, where

$$\delta_i = \|\mathscr{X} - \mathscr{T}_i\| + e_i, \tag{2}$$

for unknown error-terms e_i, $|e_i| \leq \varepsilon_i$, and unknown trajectory $\mathscr{X}$, our task is to find an approximation $\mathscr{X}'$ which minimizes the distance $\|\mathscr{X} - \mathscr{X}'\|$.

A natural measure of success of a reconstruction method is the distance $\|\mathscr{X} - \mathscr{X}'\|$. However, this distance depends on the coordinate system of the dataset, and thus tells us very little about the efficiency of the reconstruction method itself. Notice that a naïve approach to estimating $\mathscr{X}$ would be to set $\mathscr{X}'$ to the trajectory $\mathscr{T}_i$ with the smallest distance δ_i. Any meaningful method should give a solution which is closer to $\mathscr{X}$ than δ_i. Thus, we define the success-rate as

$$SR(\mathscr{X}') = 1 - \frac{\|\mathscr{X} - \mathscr{X}'\|}{\delta_{min}}, \tag{3}$$

where $\delta_{min} = \min_i(\delta_i)$ is the smallest given distance. The success-rate is 1 if the method finds $\mathscr{X}$ precisely, 0 if it returns the closest known trajectory, and finally negative if what it does is worse than just returning the closest known trajectory.

To find the unknown trajectory, we need a method which gives meaningful results, even when insufficient amount of information is given. However, the best

we can hope for, is to find a candidate trajectory which has the same properties as the properties we know about $\mathscr{X}$. If, for instance, the only information we have about $\mathscr{X}$ is that it is a car driving at an average speed of 50 km/h in Athens, then any $\mathscr{X}'$ which moves along the roads of Athens at 50 km/h is a possible solution. We thus want to minimize the difference between the given properties of $\mathscr{X}$, and the corresponding properties of the candidate $\mathscr{X}'$; in our case, the distances to the known trajectories. To this end, we define the "error" of a candidate $\mathscr{X}'$ as

$$E(\mathscr{X}') = \sum_{i=1}^{n} \big(\|\mathscr{X}' - \mathscr{T}_i\| - \delta_i\big)^2. \tag{4}$$

A natural way to solve this problem is to see it as an optimization problem, which is the essence of our method described in detail in the next section.

4 Our Method

We adopt steepest descent (gradient descent search) algorithm to find a candidate with minimum error.

The error-function (4) has value 0 exactly when the candidate trajectory is at distance δ_i to the known trajectory $\mathscr{T}_i$, for all $i \in \{1, \ldots, n\}$. Furthermore, since (4) is a positive valued function, the target trajectory is a global minimum. There may, however, be more than one global minimum, as well as several local minima; but any zero of the error-function exhausts the knowledge we can possibly have about the unknown trajectory. Recall that gradient descent algorithm finds a zero of a positive and continuously differentiable function E as follows

1. Choose a random point, x_0, in the domain of E.
2. Iteratively define $x_{i+1} = x_i - \gamma \nabla E(x_i)$, for some step-size $\gamma > 0$.
3. When $x_{i+1} = x_i$ ($\nabla E(x_i) = 0$) a (local) minimum has been reached. If $E(x_i) = 0$ we have a global minimum (since E is non-negative), and we stop. Otherwise, we restart at step 1.

The reader may notice that the success-rate as defined in Section 3, with an upper bound of 1, can be an arbitrary negative number and a lower bound for the success-rate may be hard to compute. With the gradient descent method, however, we give a lower bound on the success-rate in Theorem 1.

Theorem 1. *Any trajectory $\mathscr{X}'$ with $E(\mathscr{X}') = 0$ has success-rate*

$$SR(\mathscr{X}') \geq 1 - \frac{2\delta_{max} + \varepsilon_{max}}{\delta_{min}}, \tag{5}$$

where $\delta_{max} = \max_i(\delta_i)$ is the largest given distance, and ε_{max} is the corresponding error bound.

Proof. By the sub-additivity of the Euclidean norm, $\|\mathscr{X}' - \mathscr{X}\| \leq \|\mathscr{X}' - \mathscr{T}_i\| + \|\mathscr{T}_i - \mathscr{X}\| \leq \delta_i + (\delta_i + \varepsilon_i)$, for all $i \in \{1, \ldots, n\}$. Let $\delta_{max} = \max_i(\delta_i)$ be the largest given distance, and ε_{max} be the corresponding error bound, then $\|\mathscr{X}' - \mathscr{X}\| \leq 2\delta_{max} + \varepsilon_{max}$, and thus $SR(\mathscr{X}') \geq 1 - (2\delta_{max} + \varepsilon_{max})/\delta_{min}$. □

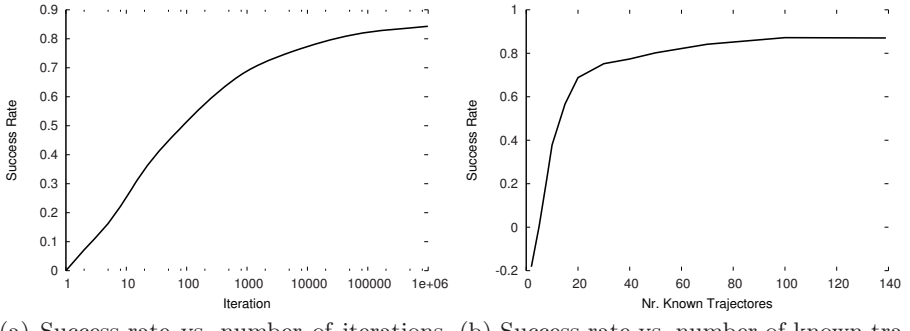
5 Experimental Results

Our reconstruction method has been tested on a dataset of GPS data from school busses in Athens[4,12]. The dataset contains 145 trajectories each with 1096 (x, y) sample points. The trajectories are recorded with samples approximately every half minute on 108 different days. For the purpose of our tests we assume that the trajectories are perfectly aligned. In all tests throughout this section, the only property used is Euclidean distance between the target trajectory and some known trajectories. No other property is known to the malicious person.

For the purpose of testing the reconstruction method described in Section 4 we implemented a limited version. In the implementation the step-size γ is set to one, and the implementation does not restart if a local maxima, or saddle point is reached. Even though time is not a primary concern in this work, we remark that it takes approximately 8 minutes to run the reconstruction method with 40 known trajectories for 50.000 iterations on a 1.7 GHz laptop on the dataset described below.

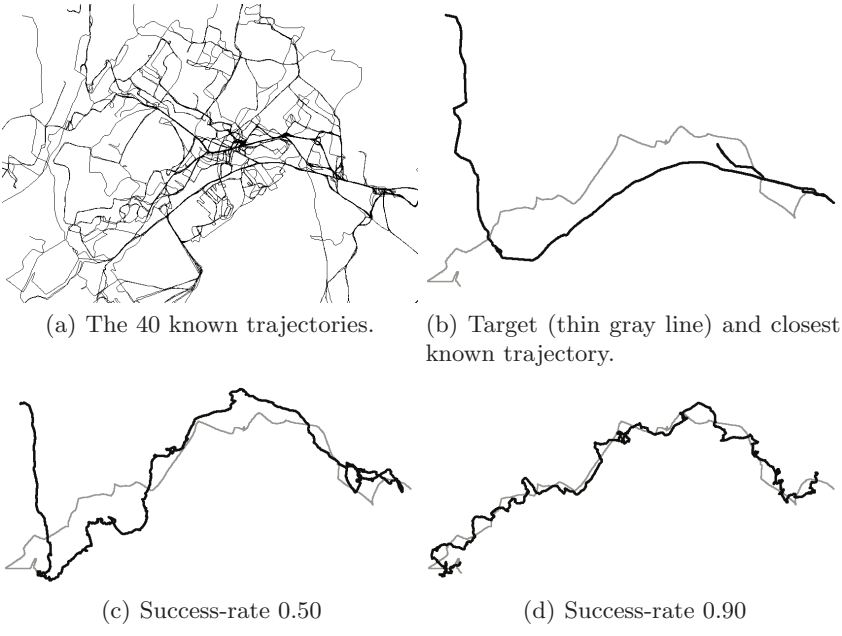
Figure 1(a) shows the convergence speed of our reconstruction method. The success-rate is an average value obtained from 15 runs of the test with 50 known trajectories, where the target trajectory is selected at random in each of the 15 runs. The x-axis shows the number of iterations in log-scale. Note that in these experiments our reconstruction method finds a candidate which is close to the best it can ever find after approximately 50.000 iterations.

Figure 1(b) shows the success-rate attainable for different numbers of known trajectories. Each sample is the average success-rate of 60 tests with 40 known trajectories, each running for 50.000 iterations. Both target and known trajectories are chosen at random in each test. The graph shows that with less than 5 known trajectories, our reconstruction method is "destructive" (the success-rate is negative); but with 8 known trajectories the success-rate grows already to 0.23. After 100 known trajectories, the success-rate stops growing.



(a) Success-rate vs. number of iterations. The x-axis is in log-scale (Average of 15 experiments with 50 known trajectories). (b) Success-rate vs. number of known trajectories (Each sample is the average of 60 experiments run for 50.000 iterations).

Fig. 1. Success-rate



(a) The 40 known trajectories.

(b) Target (thin gray line) and closest known trajectory.

(c) Success-rate 0.50

(d) Success-rate 0.90

Fig. 2. Evolution of the candidate trajectory

Figure 2 shows the evolution of a candidate in one experiment. A candidate with a success-rate of 0.9 clearly shows the whereabouts of the target. However, it must be noted that a success-rate of 0.9 may give a different visual impression for other datasets. We note that for the Athens dataset, most of the trajectories have large overlapping segments (main streets of Athens).

6 Conclusion and Future Work

In this paper we present a method for finding an unknown trajectory from knowledge of continuous properties of the trajectory. Our method is optimal in the sense that it will eventually find a candidate which exhausts all the information available about the unknown trajectory.

Our experiments show that unknown private trajectories with 1096 sample points can be reconstructed with an expected success-rate of 0.8 by knowing the distance to only 50 known trajectories. Reconstructing the trajectory perfectly with "tri-lateration" would require 2193 known trajectories.

Adding other known properties such as average speed may improve our method. Knowing the topology of the landscape in which the trajectory is lying is also likely to improve the results of our method, since many false positives will have altitudes which indicate that the candidate "moves through hills". As future work, we will investigate the effects of such properties. We assumed that noise is limited to a

known interval. A more realistic model of noise is to let the noise be chosen according to a Gaussian distribution. The present model can handle this to a certain extent using the 99.9% confidence interval as the known limited interval. However, preliminary experiments along these lines suggest that it is better to redesign the "interval function" to handle Gaussian noise.

References

1. Abul, O., Bonchi, F.: Never walk alone: Uncertainty for anonymity in moving objects databases. In: The 24th International Conference on Data Engineering (ICDE 2008) (2008)
2. Bettini, C., Mascetti, S., Wang, X.S., Jajodia, S.: Anonymity in location-based services: Towards a general framework. In: MDM, pp. 69–76 (2007)
3. First interdisciplinary workshop on mobility, data mining and privacy, rome, italy (February 2008), http://wiki.kdubiq.org/mobileDMprivacyWorkshop/
4. Frentzos, E., Gratsias, K., Pelekis, N., Theodoridis, Y.: Nearest neighbor search on moving object trajectories. In: Bauzer Medeiros, C., Egenhofer, M.J., Bertino, E. (eds.) SSTD 2005. LNCS, vol. 3633, pp. 328–345. Springer, Heidelberg (2005)
5. Giannotti, F., Nanni, M., Pinelli, F., Pedreschi, D.: Trajectory pattern mining. In: KDD 2007: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 330–339. ACM, New York (2007)
6. Gusfield, D.: Efficient methods for multiple sequence alignment with guaranteed error bounds. Bulletin of Mathematical Biology 55(1), 141–154 (1993)
7. Lee, J., Han, J., Whang, K.: Trajectory clustering: a partition-and-group framework. In: SIGMOD 2007: Proceedings of the 2007 ACM SIGMOD international conference on Management of data, pp. 593–604. ACM, New York (2007)
8. Li, X., Han, J., Lee, J.-G., Gonzalez, H.: Traffic density-based discovery of hot routes in road networks. In: Papadias, D., Zhang, D., Kollios, G. (eds.) SSTD 2007. LNCS, vol. 4605, pp. 441–459. Springer, Heidelberg (2007)
9. Mokbel, M.F., Chow, C.-Y., Aref, W.G.: The new casper: A privacy-aware location-based database server. In: ICDE, pp. 1499–1500 (2007)
10. Nanni, M., Pedreschi, D.: Time-focused clustering of trajectories of moving objects. Journal of Intelligent Information Systems 27(3), 267–289 (2006)
11. Needham, C.J., Boyle, R.D.: Performance evaluation metrics and statistics for positional tracker evaluation. In: Third International Conference on Computer Vision Systems, ICVS 2003, pp. 278–289 (2003)
12. http://www.rtreeportal.org/
13. Turgay, E.O., Pedersen, T.B., Saygın, Y., Savaş, E., Levi, A.: Disclosure risks of distance preserving data transformations. In: Ludäscher, B., Mamoulis, N. (eds.) SSDBM 2008. LNCS, vol. 5069. Springer, Heidelberg (2008)